

CLAIMS

What is claimed is:

1. A method of semaphoring between a system firmware and ACPI subsystem, the method comprising:

prior to entering a critical section by a first entity, checking a turn flag to determine if a second entity has a turn to access a critical section;

if the second entity has the turn, then checking an In flag of the second entity to determine if the second entity is in the critical section;

if the second entity is in the critical section, then waiting for the second entity to exit the critical section; and

entering the critical section by the first entity.

2. The method of claim 1, further comprising:

setting an In flag of the first entity to "TRUE" prior to entering the critical section.

3. The method of claim 1, further comprising:

setting an In flag of the first entity to "FALSE" after exiting the critical section.

4. The method of claim 1, further comprising:
if the second entity is not in the critical section,
then entering the critical section by the first entity.
5. The method of claim 1, wherein the first entity is the
system firmware and the second entity is the ACPI
subsystem.
6. The method of claim 1, wherein the first entity is the
ACPI subsystem and the second entity is the system
firmware.
7. The method of claim 1, wherein the first entity checks
a shared memory to determine priority in the critical
section.
8. An apparatus for semaphoring between a system firmware
and ACPI subsystem, the apparatus comprising:
a first entity;
a second entity;
a semaphoring system between the first entity and
second entity, wherein the first entity is configured to

check a turn flag to determine if the first entity or the second entity has a turn to access a critical section;

wherein the first entity is configured to check an In flag of the second entity to determine if the second entity is in the critical section, if the second entity has the turn, then checking an In flag;

wherein the first entity is configured to wait for the second entity to exit the critical section, if the second entity is in the critical section, and enter the critical section after exit by the second entity.

9. The apparatus of claim 8, wherein the first entity is the system firmware and the second entity is the ACPI subsystem.

10. The apparatus of claim 8, wherein the first entity is the ACPI subsystem and the second entity is the system firmware.

11. The apparatus of claim 8, wherein the first entity checks a shared memory to determine priority in the critical section.

12. An apparatus for semaphoring between a system firmware and ACPI subsystem, the apparatus comprising:

means for checking a turn flag to determine if a second entity has a turn to access a critical section, prior to entering a critical section by a first entity;

means for checking an In flag of the second entity to determine if the second entity is in the critical section, if the second entity has the turn;

means for waiting for the second entity to exit the critical section, if the second entity is in the critical section; and

means for entering the critical section by the first entity.

13. An article of manufacture, comprising:

a machine-readable medium having stored thereon instructions to:

prior to entering a critical section by a first entity, check a turn flag to determine if a second entity has a turn to access a critical section;

if the second entity has the turn, then check an In flag of the second entity to determine if the second entity is in the critical section;

if the second entity is in the critical section, then
wait for the second entity to exit the critical section;
and
enter the critical section by the first entity.